

O

PAT-NO: JP411232149A

DOCUMENT-IDENTIFIER: JP 11232149 A

TITLE: FILE COMPARISON PROCESSING METHOD AND DEVICE, AND MEDIUM
STORING FILE COMPARISON PROCESSING PROGRAM

PUBN-DATE: August 27, 1999

INVENTOR- INFORMATION:

NAME	COUNTRY
OKADA, KANAE	N/A

ASSIGNEE- INFORMATION:

NAME	COUNTRY
THE JAPAN RESERCH INSTITUTE LTD	N/A

APPL-NO: JP10046319

APPL-DATE: February 13, 1998

INT-CL (IPC): G06F012/00

ABSTRACT:

PROBLEM TO BE SOLVED: To make changeable the data of records inside two new and old files, to make addable the record of the new file, and to make eliminable the record of the old file or the like by comparing the two keys of the read two records.

SOLUTION: In the file comparison processing, the records of the new and old files are compared by using the keys (data for representing the data of the record, the data capable of identifying the record or the data used for retrieving the record) and judges whether or not the record present in the old file is 'eliminated' in the new file and whether or not the record not present in the old file is 'added' to the new file. Also, in the case that the records provided with the same key are present in the new and old files, whether or not the contents of the record are changed is judged further. That is, the keys of the new and old records read to a record area are compared, whether or not they match is judged (106), and based on the result, the output processing of updating, addition and elimination (111, 116 and 119), etc., are performed.

COPYRIGHT: (C)1999,JPO

BEST AVAILABLE COPY

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-232149

(43)公開日 平成11年(1999)8月27日

(51)Int.Cl.⁵

G 0 6 F 12/00

識別記号

5 1 0

F I

G 0 6 F 12/00

5 1 0 B

審査請求 未請求 請求項の数9 FD (全13頁)

(21)出願番号

特願平10-46319

(22)出願日

平成10年(1998)2月13日

(71)出願人 595107139

株式会社日本総合研究所

東京都千代田区一番町16番

(72)発明者 岡田 加苗

東京都千代田区一番町16番 株式会社日本
総合研究所内

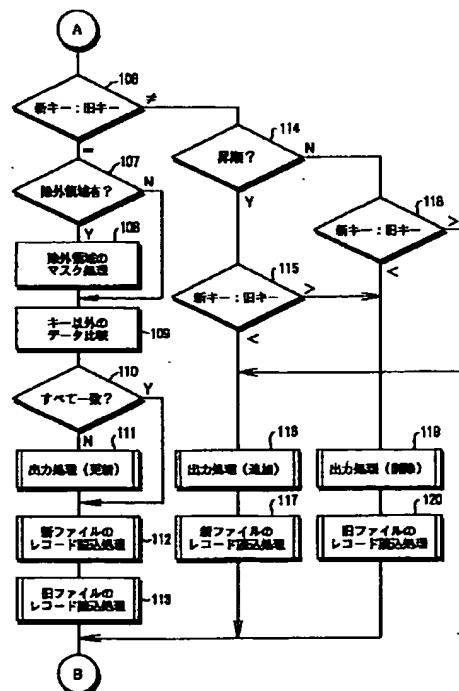
(74)代理人 弁理士 牛久 健司 (外1名)

(54)【発明の名称】 ファイル比較処理方法、装置およびファイルの比較処理プログラムを記憶した媒体

(57)【要約】

【目的】 新ファイルと旧ファイルとを比較し、レコード内容の更新のみならず、レコードの追加、削除を識別する。

【構成】 キーが昇順に配列された複数のレコードをもつ新、旧ファイルにおいて、キーの小さいレコードから順番に比較される。はじめに、新レコードと旧レコードとの間でキーが比較される。2つのキーが異なれば、次にキーの大小が比較される。新キーが旧キーよりも小さい場合、新レコードは追加されたレコードと判定され、新キーが旧キーよりも大きい場合、旧ファイルにおけるレコードが削除されたと判定される。キーが同じでかつキー以外のデータが異なれば、新レコードは更新されたレコードと判定される。キーが降順に配列された2つのファイルの比較処理も同じようにして行える。



【特許請求の範囲】

【請求項1】 第1のファイルはそれが複数の項目のデータからなる複数のレコードを含み、第2のファイルはそれが第1のファイルと同じ項目のデータからなる複数のレコードを含み、第1のファイルにおけるレコード内のデータの更新、レコードの削除または第1のファイルにないレコードの追加により生成されたものである、さらに第1のファイルと第2のファイルは、キーとして指定すべき各レコード中のデータが昇順または降順となるようにそれらのレコードが配列されている、そのような第1のファイルと第2のファイルとを比較する方法であり、第1のファイルおよび第2のファイルにおけるキーの指定を受け、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読み出し、読み出した2つのレコードの2つのキーを比較し、2つのキーが一致したときには読み出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定し、2つのキーが一致しなかったときには2つのキーの大小関係を判断し、この判断結果に応じてレコードの追加または削除を判定する、ファイル比較処理方法。

【請求項2】 データの更新と判定されたときには、第1のファイルおよび第2のファイルのそれぞれから次のレコードを読み出し、レコードの追加と判定されたときには第2のファイルから次のレコードを読み出し、レコードの削除と判定されたときには第1のファイルから次のレコードを読み出して、次のレコードのキーの比較に進む、請求項1に記載のファイル比較処理方法。

【請求項3】 第1のファイルおよび第2のファイルのキーが昇順であるか降順であるかの指定を受け、昇順であるか降順であるかの指定、および2つのキーの大小関係の判断に応じて、レコードの追加または削除を判定する、請求項1に記載のファイル比較処理方法。

【請求項4】 レコード中のデータのうち、比較処理から除外すべきデータの指定を受け、2つのキーが一致した場合に行われるレコード中のキー以外のデータの比較において、除外すべきものとして指定されたデータをマスクする、請求項1に記載のファイル比較処理方法。

【請求項5】 更新、追加または削除と判定されたレコードに関するデータを出力する、請求項1に記載のファイル比較処理方法。

【請求項6】 出力すべきレコードの範囲の指定を受け、指定された範囲のレコードに関するデータを出力する、請求項5に記載のファイルの比較処理方法。

【請求項7】 2つ以上のキーの指定を受け、読み出した2つのレコードにおいて、指定された2つ以上のすべてのキーが一致したときにデータの更新の判定処理に進み、指定された2つ以上のキーのいずれかが一致しなかったときにレコードの追加または削除の判定処理に進む、請求項1に記載のファイルの比較処理方法。

【請求項8】 第1のファイルはそれが複数の項目

のデータからなる複数のレコードを含み、第2のファイルはそれが第1のファイルと同じ項目のデータからなる複数のレコードを含み、第1のファイルにおけるレコード内のデータの更新、レコードの削除または第1のファイルにないレコードの追加により生成されたものである、さらに第1のファイルと第2のファイルは、キーとして指定すべき各レコード中のデータが昇順または降順となるようにそれらのレコードが配列されている、そのような第1のファイルと第2のファイルとを比較する装置であり、第1のファイルおよび第2のファイルにおけるキーを指定する入力装置、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読み出す手段、読み出した2つのレコードの2つのキーを比較する手段、2つのキーが一致したときには読み出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定する手段、および2つのキーが一致しなかったときには2つのキーの大小関係を判定し、この判断結果に応じてレコードの追加または削除を判定する手段、を備えたファイル比較処理装置。

【請求項9】 第1のファイルはそれが複数の項目のデータからなる複数のレコードを含み、第2のファイルはそれが第1のファイルと同じ項目のデータからなる複数のレコードを含み、第1のファイルにおけるレコード内のデータの更新、レコードの削除または第1のファイルにないレコードの追加により生成されたものである、さらに第1のファイルと第2のファイルは、キーとして指定される各レコード中のデータが昇順または降順となるようにそれらのレコードが配列されている、そのような第1のファイルと第2のファイルとを比較する処理をコンピュータに実行させるためのファイル比較処理プログラムを記憶した媒体であり、このファイル比較処理プログラムは、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読み出し、読み出した2つのレコードの2つのキーを比較し、2つのキーが一致したときには読み出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定し、2つのキーが一致しなかったときには2つのキーの大小関係を判断し、この判断結果に応じてレコードの追加または削除を判定するようにコンピュータを制御するものである、ファイル比較処理プログラムを記憶した媒体。

【発明の詳細な説明】

【0001】

【技術分野】この発明は、新旧2つのファイルを比較して、その相違点を見付けだすファイル比較処理方法および装置、ならびにファイル比較処理プログラムを格納した媒体に関する。

【0002】

【従来技術とその問題点】従来のファイル比較処理方法は、複数のレコードによって構成されたファイル（旧ファイル）と、このファイルに何らかの操作（たとえばデ

ータ変更)を施して得られるファイル(新ファイル)とをレコードごとに比較して、その一致、不一致を判断するものである。しかしながら、この従来のファイル比較処理方法は次のような問題点を持つ。

【0003】旧ファイルのレコードと新ファイルのレコードとの比較によって、2つのレコードのデータ内容に相違があれば不一致が判別されるが、どのように一致していないかまでは判別することができない。たとえば、比較の対象となった新ファイルのレコードが、旧ファイルには無いもので、新ファイルにおいて新たに追加されたレコードである場合には、従来の方法では比較された2つのレコードのデータが相互に一致しないことが判別されるに留まり、レコードが追加されたという事実までは認識することができない。

【0004】また、従来のファイル比較処理方法では、旧ファイルのレコードと新ファイルのレコードの比較は、各ファイルの並び順に行われている。各ファイルの先頭にあるレコード同士が比較され、次に2番目のレコード同士が比較される、という具合である。新ファイルが新たに追加されたレコードを含むものである場合、または旧ファイルに存在したレコードが削除されて新ファイルには存在しない場合、旧ファイルと新ファイルの対応するレコードにずれが生じ、同一のレコードが存在するにも関わらずこれを認識できない。

【0005】

【発明の開示】この発明は、新、旧2つのファイル内のレコードのデータの変更(更新)のみならず、新ファイルにおけるレコードの追加、旧ファイルのレコードの削除等も識別できるファイル比較処理方法および装置を提供するものである。

【0006】この発明はまた、上記のファイル比較処理をコンピュータに実行させるプログラムを記憶した媒体を提供するものである。

【0007】比較の対象となるファイルは2つあり、これらを第1のファイル(旧ファイル)および第2のファイル(新ファイル)とする。第1のファイルはそれぞれが複数の項目のデータからなる複数のレコードを含む。第2のファイルはそれぞれが第1のファイルと同じ項目のデータからなる複数のレコードを含み、第1のファイルにおけるレコード内のデータの更新、レコードの削除または第1のファイルにないレコードの追加により生成されたものである。さらに第1のファイルと第2のファイルは、次に言及するキーとして指定すべき各レコード中のデータが昇順または降順となるようにそれらのレコードが配列されているものである。

【0008】この発明によるファイル比較処理方法は、第1のファイルおよび第2のファイルにおけるキーの指定を受け、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読出し、読出した2つのレコードの2つのキーを比較し、2つのキーが一致したとき

には読出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定し、2つのキーが一致しなかったときには2つのキーの大小関係を判断し、この判断結果に応じてレコードの追加または削除を判定するものである。

【0009】この発明によるファイル比較処理装置は、第1のファイルおよび第2のファイルにおけるキーを指定する入力装置、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読出す手段、読出した2つのレコードの2つのキーを比較する手段、2つのキーが一致したときには読出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定する手段、および2つのキーが一致しなかったときには2つのキーの大小関係を判断し、この判断結果に応じてレコードの追加または削除を判定する手段を備えているものである。

【0010】この発明によるファイル比較処理プログラムを記憶した媒体は、第1のファイルおよび第2のファイルからレコードを1つずつ順番に読出し、読出した2つのレコードの2つのキーを比較し、2つのキーが一致したときには読出したレコード中のキー以外のデータの比較を行ってデータの更新があったかどうかを判定し、2つのキーが一致しなかったときには2つのキーの大小関係を判断し、この判断結果に応じてレコードの追加または削除を判定するようにコンピュータを制御するファイル比較処理プログラムを記憶したものである。

【0011】第1のファイルおよび第2のファイルの配列方法が昇順または降順のいずれかに定まっている場合には、定まっている配列方法にしたがってレコードの追加または削除を判定することができる。第1のファイルおよび第2のファイルのキーが昇順であるか降順であるかの指定をユーザに行わせるようにしてもよい。この場合には、昇順であるか降順であるかの指定、および2つのキーの大小関係の判断に応じて、レコードの追加または削除が判定される。

【0012】第1のファイルのレコードのキーと第2のファイルのレコードのキーが一致している場合には、レコードの存在そのものには変更がない。この場合には、レコードのキーを除くデータが比較される。データが一致すればレコードの内容にも変更がなかったのであり、データが一致しなければレコードの内容に変更(更新)があったものと判定される。

【0013】レコードのキー以外のデータの比較処理において、比較処理から除外すべきデータの指定をユーザが行えるようにしておくことが好ましい。この場合には、データ比較処理において、除外すべきものとして指定された種類のデータについてはマスクしておけばよい。

【0014】第1のファイルのレコードのキーと第2のファイルのレコードのキーとが一致しない場合には、第

2のファイルのレコードが追加されたか、第2のファイルのレコードに対応する第1のファイルのレコードが削除されたかのいずれかである。第1のファイルおよび第2のファイルのキーは大きさの順序(昇順または降順)で配列されているから、第1のファイルのレコード中のキーと第2のファイルのレコード中のキーとの大小関係を判断すれば、追加か削除かが判定できる。

【0015】このようにしてこの発明によると、2つのファイルの比較において、レコード内容が更新されたかどうかのみならず、レコードの追加および削除も判定できる。

【0016】キーは1つのみならず、2つ以上の指定を許すようにしてもよい。この場合には、読み出した2つのレコードにおいて、指定された2つ以上のすべてのキーが一致したときにデータの更新の判定処理に進み、指定された2つ以上のキーのいずれかが一致しなかったときにレコードの追加または削除の判定処理に進むようにする。

【0017】好ましい実施態様においては、更新、追加または削除と判定されたレコードに関するデータを出力する。この場合、出力すべきレコードの範囲の指定を許すようにすると一層好ましい。そして指定された範囲のレコードに関するデータのみ出力する。これは、特に大量のレコードがファイルに含まれている場合に有効である。

【0018】ファイル比較処理プログラムを記憶した媒体には、磁気ディスク、磁気テープ、光ディスク、光磁気ディスク、半導体メモリ等が含まれる。

【0019】

【実施例の説明】第1実施例

図1はファイル比較処理を行う処理装置の全体的な構成を示すものである。

【0020】ファイル比較処理を行う処理装置は、コンピュータ・システムにより実現される。このコンピュータ・システムは、たとえば大量のデータを取扱うホスト(大型)コンピュータ・システムであり、ファイル比較処理を実行するコンピュータ10を含む。コンピュータ10は、コンピュータ10により実行されるプログラムを格納するプログラム・エリア、各種演算のためのワーク・エリア、バッファ・エリアなどを提供する主記憶装置13を含む。コンピュータ10には、入力データ、コマンド等を入力するための入力装置11(キーボード、マウスなど)、データを可視的に出力するための出力装置12(CRT表示装置、印字装置など)、磁気ディスクに記録されたプログラム、データ等を読み取り、かつ磁気ディスクにデータを記録する磁気ディスク装置15、および磁気テープに記録されたデータ等を読み取り、かつ磁気テープにデータを記録する磁気テープ装置16が接続されている。

【0021】磁気ディスクには、ファイル比較処理プログラムが記憶されている。このファイル比較処理プログ

ラムは、磁気ディスク装置15によって読み出され、主記憶装置13に転送されて、そこに格納される。ファイル比較処理プログラムによってコンピュータ・システムが制御され、以下に詳述するファイル比較処理が行われる。

【0022】ファイル比較処理において比較処理の対象となるファイル(新ファイルおよび旧ファイル)は、磁気ディスク装置15の磁気ディスクに記録されているか、または磁気テープ装置16に装着される磁気テープに記録されている。磁気ディスクまたは磁気テープからファイルが読み出されて、後述する比較処理が行われる。

【0023】ファイル比較処理を行う処理装置として、パーソナル・コンピュータを用いることもできる。パーソナル・コンピュータには、一般に、フロッピー・ディスク(FD)ドライブ、CD-ROMドライブ、ハード・ディスク(HD)ドライブ等が接続されている。CD-ROMまたはFDに格納されているファイル比較処理プログラムがCD-ROMドライブまたはFDドライブによって読み出され、ハード・ディスク装置に転送され、そのハード・ディスクに格納される。ハード・ディスク装置に格納されたファイル比較処理プログラムによってパーソナル・コンピュータが制御され、ファイル比較処理が行われる。ファイル比較処理の対象となるファイルもまた、ハード・ディスク装置のハード・ディスクに格納される。

【0024】最も一般的には図1に示すコンピュータ・システムは種々のアプリケーション・プログラムをインストールすることにより、そのアプリケーションのための装置となる。ファイル比較処理プログラムはアプリケーション・プログラムの一部として利用されるユーティリティ・プログラムとして位置付けることもできる。アプリケーション・プログラムとともに、ファイル比較処理プログラムが1つの磁気ディスク、CD-ROMまたはFDに格納されている場合もありうる。

【0025】図2はファイル比較処理において比較される新ファイルと旧ファイルの一例を示すものである。

【0026】新ファイル21および旧ファイル22は表形式で表されている。表の列に付された名称(「社員番号」および「社員名」)を項目という。新ファイル21および旧ファイル22は互いに同じ項目を持つ。全項目にわたって行方向に並ぶデータは相互にリンクされている。これらのリンクされたデータの組(たとえば社員番号「0001」と社員名「トッキョ タロウ」)がレコードである。新ファイル21は5つのレコードを含み、旧ファイル22は4つのレコードを含む。

【0027】この実施例では「社員番号」は4桁の10進数表記で示されている。「社員名」はカタカナ文字で表現されている。ファイル内のデータは他の種々の形態で表記することができる。これらのデータはEBCDICコード(IBMのキャラクタ・コード)、JISコード、SJISコード、EUCコード等、表記のための任

意のコードにもとづいて表記される。

【0028】項目「社員番号」のデータは4バイトで表され、項目「社員名」のデータには10バイトの使用が許される。1レコードについて合計14バイト分の記憶場所の各位置にバイト位置番号を付す。バイト位置番号は図2の新、旧ファイル21, 22の上に1~14の数字で表されている。

【0029】ファイル比較処理ではキー（レコードのデータを代表するデータ、レコードを識別できるデータ、またはレコードを検索するのに用いるデータ）を用いて、新、旧ファイルのレコードの比較を行い、旧ファイルに存在したレコードが新ファイルでは「削除」されているか、旧ファイルには存在しなかったレコードが新ファイルに「追加」されているかが判定される。また新、旧ファイルに同じキーを持つレコードが存在する場合には、さらに進んでレコードの内容が変更なしか（「一致」）、変更されたか（「更新」）が判定される。以下に図3および図4を参照して、上述したファイル比較処理プログラムにしたがって実行されるファイル比較処理の詳細を説明する。

【0030】比較処理のためのパラメータがコンピュータ10によって読み込まれる（ステップ100）。ホスト・コンピュータの場合には、JCL（ジョブ制御言語）が設定されており、ユーザによって入力装置11から入力されたパラメータがこのJCLの中に記述されることになる。パーソナル・コンピュータの場合には、表示装置に表示されたメニュー画面にしたがってパラメータが入力装置11から入力されることになる。パラメータには次の4種類がある。

- ①KEY（キー）パラメータ
- ②SEQ（並び順）パラメータ
- ③EXP（除外）パラメータ
- ④OUTLIM（出力制限）パラメータ

【0031】①KEYパラメータは、新、旧ファイルにおいて設定すべきキーを指定するためのもので、キーとして指定すべきデータが記憶されている場所の先頭のバイト位置とキー・データの長さによって表現される。KEY=（KEYの開始位置、KEYの長さ）。社員番号をキーとする場合にはKEYパラメータは「KEY=（1, 4）」である。（1, 4）のうち、1が社員番号を表すデータの先頭のバイト位置（KEYの開始位置）であり、4がキーの長さを表す。

【0032】②SEQパラメータは、新、旧ファイルにおけるキーの配列方法を示すもので、昇順、降順のいずれかで表される。新、旧ファイルを構成する各レコードはキーを基準にあらかじめソートされている。キーが昇順にソートされている場合には、SEQパラメータは「SEQ=A」（Ascend）であり、降順にソートされている場合には、「SEQ=D」（Descend）である。

【0033】③EXPパラメータは、新、旧ファイルの

比較処理において除外したい（比較を行わない）データを指定するパラメータであり、そのデータの開始位置とそのデータの長さによって表現される。EXP=（比較処理から除外すべきデータの開始位置、該データの長さ）。具体的には図12に示すファイルに関連して後述する。

【0034】④OUTLIMパラメータは、後述する出力処理において「更新」、「追加」または「削除」と判定されたレコードのデータ内容の出力（表示および/またはプリント）を行う件数を制限するときに用いられるパラメータである。出力すべき件数は何件目（出力開始件数位置）から何件目（出力終了件数位置）というように表される。OUTLIM=（出力開始件数位置、出力終了件数位置）。OUTLIM=（1, 2）と設定された場合には、新、旧ファイルのレコードの比較処理において、「更新」、「追加」または「削除」と判定されたレコードのうち、1番目にこれらのいずれかに該当すると判定されたレコードから、2番目にこれらのいずれかに該当すると判定されたレコードまでのレコード内容および判定結果の出力が行われる。この④OUTLIMパラメータの指定のない場合には、更新、追加または削除と判定されたすべてのレコードのデータ内容が表示される。

【0035】上記①～④のパラメータのうち、この実施例では①KEYパラメータおよび②SEQパラメータ（KEY=（1, 4）、SEQ=A）が使用される。③EXPパラメータおよび④OUTLIMパラメータの使用例については他の実施例において説明する。

【0036】パラメータが妥当であるかどうかがチェックされる（ステップ101）。たとえば、図2に示す新、旧ファイル21, 22は各レコードが合計14バイト分の記憶場所を持つので、キー・パラメータが14バイト分を超える場合には（例えばKEY=（1, 20）など）、そのパラメータは妥当でないと判定される。この場合にはパラメータの訂正が要求されることになる。

【0037】パラメータの妥当性チェックが終了すると、磁気ディスクまたは磁気テープに格納されている新ファイルおよび旧ファイルが主記憶装置13に転送されて、所定のエリアに格納される（ステップ102）。比較の対象となる新、旧ファイルのすべてを主記憶装置13に転送してもよいし、適当な大きさのブロックに分け、まず最初のブロックを主記憶装置13に転送し、比較処理の進行にともなって順次ブロックごとに主記憶装置13に格納するようにしてもよい。この転送処理の多くの部分はOS（オペレーティング・システム）によって実行されるであろう。

【0038】主記憶装置13に格納された新、旧ファイル（またはブロック、以下同じ）の第1番目のレコードが主記憶装置13のワーク・エリアまたは演算論理装置内のレジスタ（以下、単に新、旧レコード・エリアという）

(どの場所に読み込まれるかは一般にOSに依存する)にそれぞれ読み込まれる(ステップ103, 104)。これらのレコード・エリアに読み込まれたレコードをそれぞれ、以下、新レコード、旧レコードということにする。また新レコードのキー・データを新キー、旧レコードのキー・データを旧キーという。

【0039】レコード・エリアに読み込まれた新、旧レコードの新キーと旧キーとが比較され、これらが一致するかどうかが判定される(ステップ106)。この判定は、新キーと旧キーのデータを、1バイト・データずつ比較することにより行われる。図8(A)に示すように、新ファイル21の第1レコードと旧ファイル22の第1レコードにおいて、それらの新キーおよび旧キーはすべてのバイトにおいて一致している(バイト位置1~4の各バイトの比較)。

【0040】新旧2つのキーが一致する場合には(ステップ106で=)，新、旧レコードのキー以外のデータが一致するかどうかが調べられる。③EXPパラメータは設定されていないので、これに関するステップ108の処理はスキップされる(ステップ107でNO)。

【0041】新、旧第1レコードの新、旧キーを除くデータの比較が行われる(ステップ109)。キー以外のデータもまた、1バイト・データごとに比較される(図8(A)のバイト位置5~12参照)。

【0042】新キーと旧キーが一致し、かつキー以外のデータが一致した場合には、2つの新、旧レコードは同一である(ステップ110でYES)。新第1レコードと旧第1レコードとは一致している。この場合にはステップ111の出力処理がスキップされ、主記憶装置13から新ファイルの次のレコード(新第2レコード)と旧ファイルの次のレコード(旧第2レコード)とが、それぞれレコード・エリアに読み込まれる(ステップ112, 113)。

【0043】新ファイル21の第2レコードと旧ファイル22の第2レコードはキーが一致しており(ステップ106で=)，かつキーを除くデータも一致している(ステップ110でYES)(図8(B)参照)。主記憶装置13から新ファイルの次のレコード(新第3レコード)および旧ファイルの次のレコード(旧第3レコード)がそれぞれレコード・エリアに読み込まれる(ステップ112, 113)。

【0044】新旧2つのキーが一致し(ステップ106で=)，かつキー以外のデータが相違している場合(ステップ110でNO)には、新レコードにおいてキー以外のデータが旧レコードのものから変更(更新)されている。新第3レコードと旧第3レコードはキー・データが一致し(ステップ106で=)，かつキーを除くデータが相違する(図8(C))。この場合には、出力処理(更新)が行われる(ステップ111)。この後、主記憶装置13から新ファイルの次のレコード(新第4レコード)お

よび旧ファイルの次のレコード(旧第4レコード)がそれぞれレコード・エリアに読み込まれる(ステップ112, 113)。出力処理(更新)の詳細は後述する。

【0045】新第4レコードのキーと旧第4レコードのキーは、図8(D)に示すように、相違している(ステップ106で≠)。読み込まれた新レコードのキーと旧レコードのキーが相違することは、新レコードが追加されたか、旧レコードが削除されたときに起こる。追加か削除かを判定するために、キーの大小関係がチェックされる。

【0046】キーの配列が昇順であるということは、ファイル内の先頭のレコードのキーが最も小さい値であり、ファイル内の末尾のレコードのキーが最も大きい値であることを意味し、キーはそれらの表す数字が順次大きくなるように配置されている。したがって、キー配列が昇順の場合、新キーが旧キーよりも大きいということは、比較の対象となっている新キーと同じキーを持つ旧レコードが削除されたことを意味する。逆に新キーが旧キーよりも小さいということは、新キーを含むレコードが新ファイルに追加された事実を表す。

【0047】キーの配列が降順の場合には、ファイル内の先頭のレコードのキーが最も大きく、ファイル内の末尾のレコードのキーが最も小さいことを意味し、キーはそれらの表す数字が順次小さくなるように配置されている。したがって、キーの配列が降順の場合、新キーが旧キーよりも大きいということは、新キーを含む新レコードが新ファイルに追加されたことを意味する。逆に新キーが旧キーよりも小さいということは、比較の対象となっている新キーと同じキーを持つ旧レコードが削除されたことを意味する。

【0048】したがって、まず新、旧レコードがキーを基準にして昇順にソートされているか、または降順にソートされているかが判定される(ステップ114)。この判定はユーザによってステップ101で入力されたSEQパラメータに基づいて行われる。この実施例では、SEQ=A(昇順)と入力されているので(ステップ114でYES)，ステップ115における新キーと旧キーの大小関係の判定へ進む。

【0049】新第4レコードのキー(0004)は旧第4レコードのキー(0005)よりも小さいので(図8(D))、ステップ115で新キー<旧キー)、新第4レコードは追加されたものと判定される。出力処理(追加)が行われた後(ステップ116)、主記憶装置13から新ファイルの次のレコード(新第5レコード)のみがレコード・エリアに読み込まれる(ステップ117)。旧ファイルの次のレコードの読み込みは行われない。既にレコード・エリアに読み込まれている旧第4レコードについての判定がまだ行われていないからである。新たにレコード・エリアに読み込まれた新第5レコードと、既にレコード・エリアに読み込まれている旧第4レコードとが次に

比較される。

【0050】新第5レコードのキー(0006)は旧第4レコードのキー(0005)よりも大きい(図8(E), ステップ115で新キー>旧キー)。旧第4レコードは新ファイルに存在せず、削除されたものと判定される。この場合には出力処理(削除)が行われた後(ステップ119), 旧ファイルの次のレコードの読み込みのみが行われる(ステップ120)。新ファイルの次のレコードの読み込みは行われない。なぜなら、新第5レコードについての判定がまだ行われていないからである。

【0051】この実施例では旧ファイル22内には4個のレコードしかないので、旧ファイル22には次にレコード・エリアに読み込むべきレコードは存在しない。このような場合には、後述するように、旧ファイルのレコードのキーとして、とりうる最大数(4桁の10進数表記では「9999」で表現される、16進数でいえばFFF...、2進数なら11111...である)が用いられる。

【0052】したがって、新第5キー(0006)と最大値(9999)が比較される。これらは一致しないので(ステップ106で≠), その大小関係が判定され、結局新第5レコードは追加されたものと判定される(ステップ114でYES, ステップ115で新キー<旧キー)。出力処理(追加)ののち(ステップ116), 新ファイルのレコードの読み込みが行われる(ステップ117)。新ファイルは5個のレコードしか含まないから、新ファイルの次のレコードのキーとして最大値(9999)が用いられる。

【0053】新ファイルおよび旧ファイルのキーが、ともに最大値になると、新、旧ファイルのいずれにも、比較すべきレコードはもはや存在しないと判定される(ステップ105でYES)。図10に示す件数表が作成されて出力される(ステップ121)。これによりファイル比較処理は終了する。件数表には新、旧レコードの件数、更新レコードの件数、追加レコードの件数および削除レコードの件数がまとめられる。この件数表は実際に表示またはプリントされるときには、図示のように漢字で表されるのではなく、記号等を用いて表現されよう。一般的には枠も設けられないであろう。

【0054】新ファイルのレコードの読み込み処理(ステップ103, 112, 117)(図5)および旧ファイルのレコードの読み込み処理(ステップ104, 113, 120)(図6)について説明する。

【0055】新レコード・エリアは新キーを格納する新キー・エリアとそれ以外のデータを格納する新データ・エリアとを持つ。旧レコード・エリアは旧キーを格納する旧キー・エリアとそれ以外のデータを格納する旧データ・エリアとを持つ。

【0056】図5において、主記憶装置13から新ファイルの1つのレコード(新レコード)が読み出され、その

新レコードの新キーが新キー・エリアに、新キー以外のデータが新データ・エリアにそれぞれ格納される(ステップ203)。

【0057】新ファイル内のすべてのレコードの読み出しが終了したときには(ステップ201でYES), 新キー・エリアに最大値が格納される(ステップ204)。

【0058】図6に示す旧ファイルのレコードの読み込み処理においても、旧キー・エリアへの旧キーの格納および旧データ・エリアへの旧キー以外のデータの格納が同じ10ようにして行われる。

【0059】図4に示すステップ106, 115 および118の新キーと旧キーの比較処理は、新キー・エリアの内容と旧キー・エリアの内容とを比較するものである。ステップ109, 110 のデータの比較処理では、新データ・エリアの内容と旧データ・エリアの内容との比較が行われる。ステップ105の判定では新、旧キー・エリアの内容が参照される。

【0060】図4において、新キーと旧キーとの比較(ステップ115, 118)の前に昇順か降順かを判定して20いるが(ステップ114), まず新キーと旧キーとの大小関係を判定し、その後に昇順か、降順かを調べて、追加(ステップ116)または削除(ステップ119)を判定してもよい。

【0061】図7は出力処理(ステップ111, 116, 119)の詳細を示すものである。

【0062】この実施例では④OUTLIMパラメータは設定されていない(ステップ401でNO)。新レコードが更新されたものと判定された場合には(出力処理(更新))(ステップ111), 更新である旨、および更新されたレコード(新レコード)および更新前のレコード(旧レコード)が出力される(ステップ402)(図9の第1行目参照)。社員名は1バイトずつ比較され、更新のあった社員名には1バイトごとに*(アスタリスク)が付される。新、旧ファイルにおいてそれらが第何番目のレコードであるのか、も出力される。

【0063】新レコードが追加されたものと判定された場合には(出力処理(追加))(ステップ116), 追加である旨、追加されたレコード(新レコード)が新ファイルにおいて第何番目のレコードであるか、および追加された新レコードが出力される(ステップ402)(図9の第2行目および第4行目参照)。

【0064】旧レコードが削除されたものと判定された場合には(出力処理(削除))(ステップ119), 削除である旨、削除されたレコード(旧レコード)が旧ファイルにおいて第何番目のレコードであるか、および削除された旧レコードが出力される(ステップ402)(図9の第3行目参照)。図9に示す出力データもまた実際の表示、プリントでは記号等を用いて表現されよう。枠も一般的には設けられないであろう。

【0065】更新と判定されたレコード数、追加と判定

された数および削除と判定されたレコード数をそれぞれ計数するカウンタのうち、該当するカウンタがインクリメントされる（ステップ403）。ステップ404～405の処理については、後述する。

【0066】件数表の作成処理（図3、ステップ121）では、ここで計数された数値が更新レコードの件数、追加レコードの件数、削除レコードの件数としてそれぞれ出力される。

【0067】第2実施例

第2実施例における新、旧ファイルが図11に示されている。これらの新、旧ファイル23、24ではキーが降順で配列されている。レコードの内容そのものは図2に示すものと同じである。図1に示す処理装置の構成、図3～図7に示すフローチャートが第2実施例においてもそのまま適用される。

【0068】①KEYパラメータとして（KEY=（1, 4））が設定される。②SEQパラメータとしては、SEQ=D（降順）が設定される（図3、ステップ101）。より大きいキーを持つ新、旧レコードから順に比較処理が行われる。主記憶装置13からレコード・エリアに読込まれた新、旧レコードの新キーと旧キーとの比較が行なわれる。新、旧キーが等しい場合には（図4、ステップ106 で=），ステップ107、または109 以降の処理に進む。新、旧キーが等しくない場合には（ステップ106 で≠），SEQパラメータがチェックされ（ステップ114），降順であるからステップ118 に進み、新、旧キーの大小関係が判別される。上述のように、新キーが旧キーよりも大きい場合には新キーを含む新レコードが新ファイルに追加されたと判定され、新キーが旧キーよりも小さい場合には、比較の対象となっている新キーと同じキーを持つ旧レコードが削除されたと判定される。この後、出力処理（ステップ116、119）および次のレコード（新レコードまたは旧レコード）の読込処理に進む（ステップ117、120）。

【0069】第3実施例

第3実施例は、①KEYパラメータおよび②SEQパラメータに加えて、③EXPパラメータを用いる例である。図12は第3実施例において比較される新ファイルと旧ファイルの一例を示すものである。新ファイル25、26は図2に示す新、旧ファイル21、22のバイト位置15～18に項目「参照月日」のデータが加えられているものである。KEYパラメータおよびSEQパラメータとしては第1実施例と同じようにKEY=（1, 4）およびSEQ=Aが入力される。上述のように比較処理の除外データを設定するEXPパラメータは、EXP=（比較処理から除外したいデータの開始処理、該データの長さ）のように表される。参照月日のデータを除外するので、入力されるEXPパラメータは（15, 4）となる。

【0070】第3実施例においても図1に示す処理装置の構成、図3～図7に示すフローチャートがそのまま用

いられる。第1実施例と異なる点は、EXPパラメータが設定されているので、図4のステップ107 でYESとなることである。

【0071】新、旧レコードのキー以外のデータについて、EXPパラメータにより設定された除外すべきデータがマスクされた上で（ステップ108），比較処理が行われる（ステップ109）。社員番号（キー）と参照月日のデータを除くデータ、すなわち社員名のデータについてのみ比較処理が行われる。

【0072】新ファイル25と旧ファイル26との比較処理において、参照月日のみが異なる新、旧レコード（たとえば新ファイル25の第1レコードと旧ファイル26の第1レコード）が比較されると、参照月日のデータが異なるので新ファイル25の第1レコードは「更新」と判定されてしまう。項目「参照月日」についてEXPパラメータを設定しておくことによって、参照月日のみ異なる新、旧レコードを、一致するレコードとして取り扱うことができる。

【0073】第4実施例

第4実施例は、①KEYパラメータ、②SEQパラメータおよび③EXPパラメータに加えて、④OUTLIMパラメータを用いる例である。図12に示す第3実施例の新、旧ファイル25、26が用いられる。図1に示す処理装置の構成、および図3～図7のフローチャートがそのまま利用される。

【0074】KEYパラメータ、SEQパラメータおよびEXPパラメータはそれぞれKEY=（1, 4）、SEQ=A、EXP=（15, 4）と設定されているものとする。OUTLIMパラメータとしてOUTLIM=（1, 2）と入力されているものとする。

【0075】第4実施例では図7に示すステップ401の出力制限の判断でYESとなり、ステップ404の処理に進む。

【0076】出力処理は上述のように、更新、追加または削除と判定されたレコードについてのみ行なわれる（ステップ111、116、119）。一方、OUTLIMパラメータは、詳細出力すべきレコードを何件目（出力開始件数位置）から何件目（出力終了件数位置）というように指定するものである。更新、追加または削除と判定されたレコードがこのOUTLIMパラメータによって指定された範囲内に入っているかどうかが判定される（ステップ404）。この判定は「更新」と判定されたレコード数を計数するカウンタ、「追加」と判定されたレコード数を計数するカウンタおよび「削除」と判定されたレコード数を計数するカウンタの計数値の和がOUTLIMパラメータによって指定される範囲内に入っているかどうかに基づいて行なわれる。この範囲内に入っていれば（ステップ404 でYES），そのレコードの詳細が列出される（ステップ405）。入っていないければ（ステップ404 でNO），レコードの詳細列出は行われない。

い。この後、上記のカウンタのうちのいずれかに該当するものがインクリメントされる(ステップ403)。

【0077】OUTLIM=(1, 2)と設定された場合には、図9に示す出力リストのうちの第1行目および第2行目が出力されることになる。このOUTLIMパラメータは新、旧ファイルが膨大な数のレコードを含む場合に、その一部だけを抽出して出力させるときに有効である。

【0078】第5実施例

KEYパラメータおよびEXPパラメータとして、それぞれ複数種類のデータを設定することもできる。第5実施例はKEYパラメータおよびEXPパラメータとして、それぞれ2種類のデータを指定した例である。図13は新、旧ファイルの例を示す。新ファイル27および28は「社員番号」、「社員名」、「所得資格コード」、「参照月日」および「更新月日」の項目をもつ。図2に示す新、旧ファイル21, 22と比較すると、バイト位置15, 16に「資格取得コード」のデータが、バイト位置17~20に「参照月日」のデータが、バイト位置21~24に「更新月日」のデータがそれぞれ加えられている。

【0079】社員番号および取得資格コードがそれぞれキーとして設定されるものとする。社員番号を第1キー、取得資格コードを第2キーとする。第1キーは昇順で配列されている。第2キーは2つ以上のレコードにおいて第1キーが同じである場合(たとえば新ファイル27において、社員番号0001のレコードが2つある)に効果的に用いられる。第2キーは第1キーが同一である複数のレコードにおいて昇順に配列されている。

【0080】図1に示す処理装置、および図3、図5~図7のフローチャートがそのまま用いられる。図4のフローチャートの一部が第1実施例と異なり、この異なる部分のみが図14に示されている。

【0081】パラメータの入力において(図3、ステップ101)、2つのKEYパラメータ、KEY=(1, 4)(「社員番号」とKEYパラメータ(15, 2)(「資格取得コード」)が入力される。また、2つのEXPパラメータとしてEXP=(17, 4)(「参照月日」)およびEXP=(21, 4)(「更新月日」)が入力される。SEQパラメータはSEQ=Aである。

【0082】新、旧ファイル27, 28において、第1キー(項目「社員番号」のデータ)の小さい順に、新、旧レコードがそれぞれ新、旧レコード・エリアに読み込まれる。新レコードの第1キー(新第1キー)と、旧レコードの第1キー(旧第1キー)とが比較され、これらが同じである場合には(ステップ131で=)、読みて、新レコードの第2のキー(新第2キー)と旧レコードの第2キー(旧第2キー)とが比較される(ステップ132)。

【0083】新第2キーと旧第2キーとが一致したときには(ステップ132で=)、③EXPパラメータの設定の有無が判定される(ステップ107)。

【0084】この実施例では項目「参照月日」および「更新月日」が除外されているから、これらの項目のデータがマスクされた上で(ステップ108)、新、旧レコードにおける第1キーおよび第2キーを除く他のデータの比較処理が行われる。(ステップ109)。

【0085】一方、新第1キーと旧第1キーとが異なる場合(ステップ131で≠)、新第1キーと旧第1キーの大小関係が判定される(ステップ114でYES、ステップ115)。新第1キーと旧レコードの第1キーとが同じで(ステップ131で=)、かつ新第2キーと旧第2キーとが異なる場合には(ステップ132で≠)、新第2キーと旧第2キーの大小関係が判定される(ステップ114でYES、ステップ115)。たとえば、新、旧ファイル27, 28の新第1レコードと旧第1レコードにおいて、新第1キーと旧第1キーは同じであり((0001)と(0001))、かつ新第2キーと旧第2キーは相違する((01)と(02))。この場合、ステップ115において新第2キーと旧第2キーの大小関係が判定される。これらの大小関係の判定結果にしたがって、追加または削除の判定に関する処理に進む。図13に示す新ファイル27の第1レコードは追加されたものと判定されることになる。

【0086】このようにキーおよび比較除外データとして、それぞれ2種類のデータを指定することができる。もちろん、3つ以上の種類のデータをキーまたは除外データとしてそれぞれ設定することができるのいうまでもない。

【0087】上記実施例では社員番号や取得資格コードのような数字がキーとして用いられているが、文字、記号等もキーとして用いることができる。たとえば、社員名(氏のみ、または氏名)をキーとして用いることができる。社員名を表すキャラクタ・コードは最終的には数字(2進数、10進数、16進数)で表現されるので、それらの間に大小関係があり、昇順または降順に配列することができるからである。また、複数の項目の一部または全部を1つのキーとしてもよい。たとえば、社員番号の全部と社員名の第一文字とによってキーを構成することもできる。

【0088】さらに、上述の実施例では4種類のパラメータがユーザーによって入力されるが、パラメータの種類をさらに増やすこともできる。たとえば、16進数または文字(キャラクタ)を出力データの表現形態とすることを、パラメータにより指定することができる。

【図面の簡単な説明】

【図1】ファイル比較処理を行う処理装置の全体的構成を示すブロック図である。

【図2】第1実施例の新ファイルおよび旧ファイルを示す。

【図3】ファイル比較処理の処理手順を示すフローチャートである。

17

【図4】ファイル比較処理の処理手順を示すフローチャートである。

【図5】新ファイルのレコードの読み処理の手順を示すフローチャートである。

【図6】旧ファイルのレコードの読み処理の手順を示すフローチャートである。

【図7】出力処理の処理手順を示すフローチャートである。

【図8】(A)～(F)は図2に示す新ファイルと旧ファイルのレコードを比較している状態を示す。

【図9】図2に示す新ファイルと旧ファイルの比較結果の詳細な出力の例を示す。

【図10】図2に示す新ファイルと旧ファイルの比較処理により得られる件数表の例を示す。

【図11】第2実施例における新ファイルおよび旧ファイルの例を示す。

18

【図12】第3および第4実施例の新ファイルおよび旧ファイルを示す。

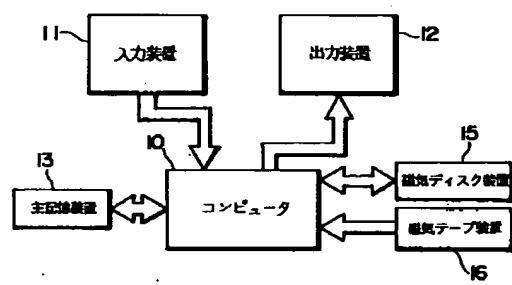
【図13】第5実施例の新ファイルおよび旧ファイルを示す。

【図14】第5実施例のファイル比較処理の処理手順を示すフローチャートの一部を示すものであり、図4のフローチャートの一部を変更するものである。

【符号の説明】

10	コンピュータ
11	入力装置
12	出力装置
13	主記憶装置
14	新ファイル
15	磁気ディスク装置
16	磁気テープ装置
21	新レコード件数
22	旧レコード件数
23	更新レコード件数
24	追加レコード件数
25	削除レコード件数
26	新ファイル
27	旧ファイル

【図1】



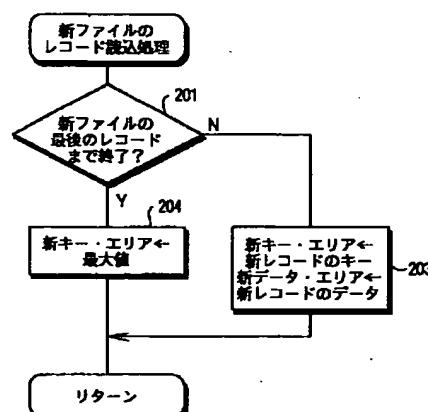
【図2】

新ファイル														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	
社員番号	社員名													
0 0 0 1	トッキヨ タロウ													
0 0 0 2	ジツヨウ ジロウ													
0 0 0 3	イショウ ハナコ													
0 0 0 4	ショウヒヨウ ハナエ													
0 0 0 6	ハツメイ サブロウ													

【図10】

新レコード件数	5
旧レコード件数	4
更新レコード件数	1
追加レコード件数	2
削除レコード件数	1

【図5】

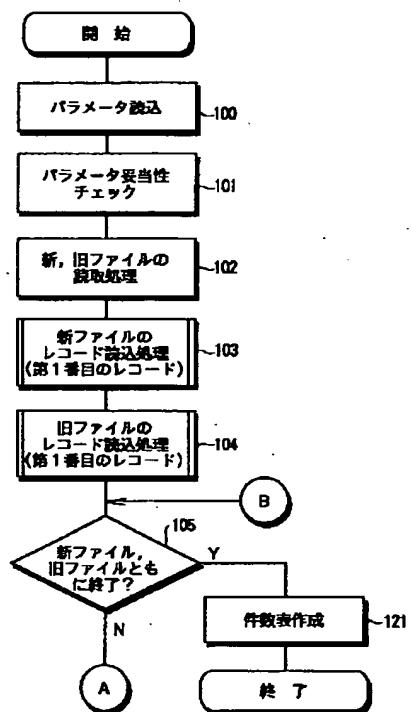


旧ファイル														
1	2	3	4	5	6	7	8	9	10	11	12	13	14	
社員番号	社員名													
0 0 0 1	トッキヨ タロウ													
0 0 0 2	ジツヨウ ジロウ													
0 0 0 3	コウアン ハナコ													
0 0 0 5	ヤクム ハナ													

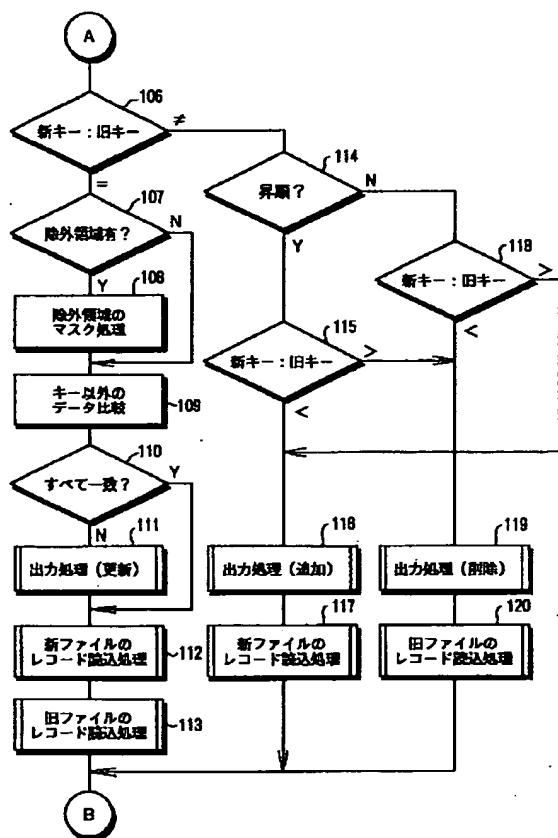
【図9】

更新	新第3レコード	0 0 0 3	イ	シ	ヨ	ウ	ハ	ナ	コ
	旧第3レコード	0 0 0 3	*	*	*	*	*	*	*
追加	新第4レコード	0 0 0 4	シ	ョ	ウ	ヒ	ヨ	ウ	ハ
	旧第4レコード	0 0 0 5	ヤ	ク	ム	ハ	ナ	エ	
削除	新第5レコード	0 0 0 6	ハ	ツ	メ	イ	サ	ブ	ロ
	旧第5レコード								

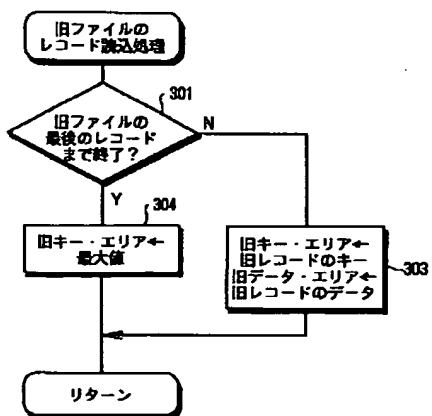
【図3】



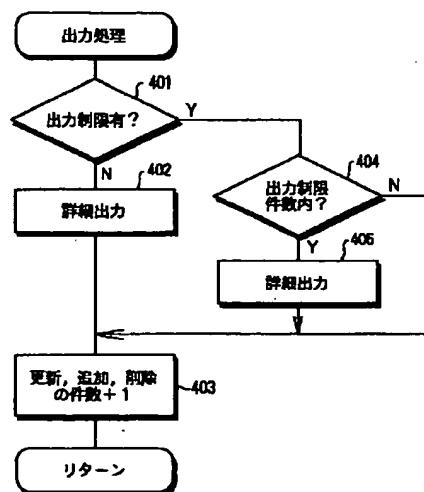
【図4】



【図6】



【図7】



【図8】

	1	2	3	4	5	6	7	8	9	10	11	12
(A)	新第1レコード	0	0	0	1	ト	ッ	キ	ヨ	タ	ロ	ウ
	旧第1レコード	0	0	0	1	ト	ッ	キ	ヨ	タ	ロ	ウ
						一致				一致		
(B)	新第2レコード	0	0	0	2	ジ	ッ	ヨ	ウ	ジ	ロ	ウ
	旧第2レコード	0	0	0	2	ジ	ッ	ヨ	ウ	ジ	ロ	ウ
						一致				一致		
(C)	新第3レコード	0	0	0	3	イ	シ	ョ	ウ	ハ	ナ	コ
	旧第3レコード	0	0	0	3	コ	ウ	ア	ン	ハ	ナ	コ
						一致				相違		
(D)	新第4レコード	0	0	0	4							
	旧第4レコード	0	0	0	5							
						相違	⇒	0004<0005 (追加)				
(E)	新第5レコード	0	0	0	6							
	旧第4レコード	0	0	0	5							
						相違	⇒	0006>0005 (削除)				
(F)	新第5レコード	0	0	0	6							
						9	9	9	9			
						相違	⇒	0006<9999 (追加)				

【図11】

新ファイル													
社員番号	社員名												
0 0 0 6	ハ	ツ	メ	イ	サ	ブ	ロ	ウ					
0 0 0 4	ショウ	ヒ	ヨウ	ハ	ナ	エ							
0 0 0 3	イ	シ	ョウ	ハ	ナ	コ							
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ							
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ						

23

旧ファイル													
社員番号	社員名												
0 0 0 5	ヤ	ク	ム	ハ	ナ								
0 0 0 3	コ	ウ	ア	ン	ハ	ナ	コ						
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ							
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ						

24

【図12】

新ファイル																
社員番号	社員名															
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ	0	6	0	1	0	5	2	5	
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ		0	6	0	1	0	5	2	5	
0 0 0 3	イ	シ	ョウ	ハ	ナ	コ		0	6	0	1	0	5	2	5	
0 0 0 4	ショウ	ヒ	ヨウ	ハ	ナ	エ		0	6	0	1	0	5	2	5	
0 0 0 6	ハ	ツ	メ	イ	サ	ブ	ロ	ウ	0	6	0	1	0	5	2	5

25

旧ファイル

旧ファイル															
社員番号	社員名														
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ	0	5	0	1	0	4	2	5
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ		0	5	0	1	0	4	2	5
0 0 0 3	コ	ウ	ア	ン	ハ	ナ	コ	0	5	0	1	0	4	2	5
0 0 0 5	ヤ	ク	ム	ハ	ナ			0	5	0	1	0	4	2	5

26

新ファイル

新ファイル															
社員番号	社員名														
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ	0	5	0	1	0	4	2	5
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ		0	5	0	1	0	4	2	5
0 0 0 3	コ	ウ	ア	ン	ハ	ナ	コ	0	5	0	1	0	4	2	5

27

旧ファイル

旧ファイル															
社員番号	社員名														
0 0 0 1	ト	ッ	キ	ヨ	タ	ロ	ウ	0	5	0	1	0	4	2	5
0 0 0 2	ジ	ッ	ヨウ	ジ	ロ	ウ		0	5	0	1	0	4	2	5
0 0 0 3	コ	ウ	ア	ン	ハ	ナ	コ	0	5	0	1	0	4	2	5

28

【図14】

